

**TRIBHUVAN UNIVERSITY  
INSTITUTE OF ENGINEERING**

PASCHIMANCHAL CAMPUS  
LAMACHAUR, POKHARA



**SOFTWARE REQUIREMENT SPECIFICATIONS**

**on**

**ePurano**

**BY**

SEAMOON PANDEY - PAS078BCT035  
RAKSHYA RAUT- PAS078BCT028

**TO**

**Department of Computer and Electronics Engineering  
POKHARA, NEPAL**

June 8, 2024

# Table of Contents

<b>Chapter 1: Introduction.....</b>	<b>1</b>
1.1 Background Theory.....	1
1.2 Purpose .....	1
1.3 Scope.....	1
1.4 Overview .....	1
<b>Chapter 2: Overall Description .....</b>	<b>3</b>
2.1 Product Perspective .....	3
2.2 Product Functions.....	3
2.2.1 User.....	3
2.2.2 Administrator .....	3
<b>Chapter 3: Specific Requirements .....</b>	<b>4</b>
3.1 Interface Requirements.....	4
3.1.1 User Interfaces: .....	4
3.1.2 Admin Interfaces:.....	4
3.1.3 External Interfaces:.....	4
3.2 Functional Requirements.....	4
3.2.1 User Registration and Authentication: .....	5
3.2.2 User Profile:.....	5
3.2.3 Listing items for Sale:.....	5
3.2.4 Browsing and Searching: .....	5
3.2.5 Payment Integration: .....	5
3.3 Non-Functional Requirements: .....	5
3.3.1 Performance:.....	5
3.3.2 Scalability: .....	6
3.3.3 Reliability: .....	6
3.3.4 Security: .....	6
3.3.5 Usability:.....	6
3.3.6 Maintainability:.....	6
<b>Abbreviations.....</b>	<b>7</b>

# Chapter 1: Introduction

## 1.1 Background Theory

The ePurano application is designed as a platform to facilitate peer-to-peer transactions of secondhand items. Recognizing the growing demand for sustainable and cost-effective alternatives to purchasing new products, ePurano aims to create a seamless, user-friendly experience for both buyers and sellers in the secondhand market. The application will provide functionalities such as item listings, user profiles, secure messaging, payment processing, and review systems to ensure trust and transparency between users.

This Software Requirements Specification (SRS) document outlines the comprehensive requirements for the development of ePurano, detailing functional and non-functional requirements, system architecture, user interfaces, and the technological stack. The objective is to ensure a clear understanding of the application's capabilities, constraints, and performance expectations, providing a solid foundation for the development and deployment phases.

ePurano's mission is to empower individuals to efficiently buy and sell pre-owned goods, fostering a community-driven marketplace that promotes sustainability and economic efficiency. This document serves as a roadmap to guide the project team in delivering a robust and reliable platform that meets the needs of its users while adhering to industry standards and best practices.

## 1.2 Purpose

The purpose of this Software Requirements Specification (SRS) document is to outline the requirements for the development of ePurano, a platform designed to facilitate peer-to-peer transactions of secondhand items. This document aims to ensure a clear understanding of the application's capabilities, constraints, and performance expectations, providing a solid foundation for the development and deployment phases.

## 1.3 Scope

ePurano is an application platform that enables users to buy and sell secondhand items in a seamless, user-friendly environment. The application will provide functionalities such as item listings, user profiles, secure messaging, payment processing, and review systems to ensure trust and transparency between users. The goal is to empower individuals to efficiently buy and sell pre-owned goods, fostering a community-driven marketplace that promotes sustainability and economic efficiency.

## 1.4 Overview

The ePurano platform will be developed using the Waterfall software development model, a linear and sequential approach ideal for projects with well-defined requirements. This model ensures systematic execution through the following phases:

- **Requirements Analysis** Gather and document all requirements for the ePurano platform.
- **System Design** Outline the architecture and detailed design based on the requirements.
- **Implementation** Develop the platform according to the design specifications.
- **Integration and Testing** Conduct thorough testing to ensure functionality and reliability.

Waterfall model follows a sequential approach, with each phase of the SDLC completed before moving to the next phase.

# Chapter 2: Overall Description

## 2.1 Product Perspective

ePurano is a new application designed to create a marketplace for peer-to-peer transactions of secondhand items. Unlike traditional e-commerce platforms, ePurano focuses specifically on pre-owned goods, promoting sustainability and cost savings for users. The platform will be standalone but will integrate with various third-party services for payment processing, user authentication, and messaging.

## 2.2 Product Functions

The primary functions of ePurano include:

### 2.2.1 User

- 

### 2.2.2 Administrator

# Chapter 3: Specific Requirements

## 3.1 Interface Requirements

Interface requirements define how the system interacts with users and other systems. For eprano, these would include:

### 3.1.1 User Interfaces:

- **Homepage:** A clean, attractive homepage with links to major sections such as categories, search bar, user login/register, and featured listings.
- **Listing Page:** Detailed view of an item, including photos, description, price, seller information, and contact options.
- **User Dashboard:** Access to user's listings, messages, profile settings, and transaction history.
- **Search Interface:** A user-friendly search bar with filters and sorting options.

### 3.1.2 Admin Interfaces:

- **Moderation Tools:** Interface for administrators to manage user reports, edit or remove inappropriate listings, and handle user bans.
- **Analytics Dashboard:** Overview of platform metrics such as user activity, listings, and transaction volumes.

### 3.1.3 External Interfaces:

- **Payment Gateway:** Secure integration with third-party payment processors.
- **Email Service:** Integration with an email service provider for sending notifications, password resets, and other communications.
- **Social Media Integration:** Options for users to log in using their social media accounts and share listings on social media platforms.

## 3.2 Functional Requirements

Interface requirements define how the system interacts with users and other systems.

### **3.2.1 User Registration and Authentication:**

- Users must be able to create accounts using email or social media logins.
- must be able to log in and log out securely.
- Users should be able to reset their passwords if forgotten.

### **3.2.2 User Profile:**

- Users should have profiles where they can manage their personal information and preferences.
- Users should be able to view their own buying and selling history.

### **3.2.3 Listing items for Sale:**

- Users must be able to create new listings, including adding photos, descriptions, categories, and pricing.
- Users should be able to edit or delete their listings.
- Users should be able to edit or delete their listings.
- Users should be able to mark items as sold.

### **3.2.4 Browsing and Searching:**

- Users must be able to browse items by categories.
- Users should be able to use a search function to find specific items using keywords, filters (price range, category, condition), and sort options (newest, price, etc.).

### **3.2.5 Payment Integration:**

- The platform should support secure payment methods for online transactions.
- Users should be able to track the status of their payments.

## **3.3 Non-Functional Requirements:**

Non-functional requirements define the system's operation rather than specific behaviors.

### **3.3.1 Performance:**

- The website should load within 3 seconds on a standard broadband connection.
- The system should handle at least 10,000 concurrent users without performance degradation.

### **3.3.2 Scalability:**

- The platform should be able to scale to accommodate growing numbers of users and listings.

### **3.3.3 Reliability:**

- Data backups should be performed daily to prevent data loss.

### **3.3.4 Security:**

- User data should be encrypted during transmission and storage.

### **3.3.5 Usability:**

- The user interface should be intuitive and easy to navigate.
- Mobile responsiveness is essential to ensure usability on various devices, including smartphones and tablets.

### **3.3.6 Maintainability:**

- The codebase should be well-documented to facilitate future maintenance and updates.
- The system should support easy updates and feature additions without major downtime.



# Abbreviations

**ePurano** The name of the application platform

**P2P** Peer to peer

**SRS** Software Requirements Specification

**SDLC** Software Development Life Cycle